

# Адаптивный алгоритм трансформации спутниковых изображений, содержащих разрывы и помехи приема в географические проекции с частичным восстановлением сбоев.

Непомнящий П.В.  
ЦОС и ВТ МФТИ  
Москва, Россия  
npv@cos.ru

## Аннотация

Описан переработанный алгоритм трансформации растровых изображений в соответствии с заданным произвольным преобразованием. Путем введения матрицы покрытия удалось избежать ограничения на непрерывность исходных данных. Алгоритм состоит из двух этапов – предобработки и аппроксимации. На этапе предобработки строятся матрицы распределения и покрытия. На этапе аппроксимации с использованием данных из матриц аппроксимируются пиксели результирующего изображения с адаптивным выбором наилучшего метода аппроксимации. Аппроксимация происходит по исходной пиксельной сетке.

**Ключевые слова:** преобразование изображений, трансформация, географические проекции, геоинформационные системы, NOAA, Terra, Aqua, EOS-AM1, EOS-PM1, MODIS, AVHRR, bow-tie effect.

## 1. ПЕРВОНАЧАЛЬНЫЙ АЛГОРИТМ

Задача трансформации изображения состоит в том, что требуется перевести исходное изображение, привязанное к какой-либо системе координат, в другую систему координат, для чего задается функция, осуществляющая преобразование точки из исходной системы координат (ИСК) в результирующую (РСК). Ввиду того, что изображения растровые, требуется аппроксимация пикселей выходного изображения по пикселям входного [1]. Для того, чтобы обеспечить минимальную потерю точности, необходимо, чтобы преобразование осуществлялось за один проход (без промежуточных результатов) и чтобы использовались наиболее точные методы аппроксимации.

У первоначально разработанного алгоритма преобразования изображений [2,3] имелось несколько существенных недостатков. Это:

- требование к отсутствию разрывов на исходных данных (сетка исходных координат должна быть равномерной) при преобразовании;
- интерполяция по выходной, а не по входной координатной сетке;
- плохая расширяемость алгоритма (алгоритм писался конкретно для целей проецирования, и его использование для других преобразований было неудобно);
- невозможность задания области определения функции, задающей преобразование.

Требование к отсутствию разрывов было препятствием при проецировании, например, экватора, по которому идет раз-

рыв в проекции поперечного Меркатора. Причина этого требования была в том, что для определения области, на которую попадет исходное изображение после преобразования, на выходном изображении формировался многоугольник, полученный путем проецирования границ исходной области на выходное изображение. Запрет на разрывы выходит из запрета на самопересечения данного многоугольника. В случае наличия самопересечений данный многоугольник теряет смысл как граница области, куда ляжет после преобразования исходное изображение. Наиболее ярко данные проблемы возникали при проецировании мировых карт, в ситуациях, когда нельзя избежать появления самопересечений. Такие проблемы приходилось решать, например, проецируя мировую карту по частям. Мировая карта в поперечном Меркаторе проецировалась за два раза – сначала северное, а потом южное полушарие, при этом вдоль экватора наблюдались визуальные дефекты на выходном изображении.

Также возникла необходимость обработки сеансов поступающих со спутников Terra (EOS-AM1), Aqua (EOS-PM1), прибор MODIS. Основные ссылки по аппаратуре, приему в режиме Direct Broadcast можно найти на официальном сайте [4], детальное описание прибора MODIS приводится в [5]. Особенность этих сеансов в том, что они состоят из сканов (полос высотой в 10, 20 или 40 пикселей, в зависимости от пространственного разрешения спектрального канала). Один скан – это один проход сканирующей линейки элементов, направленной к земле и совершающей повороты для сканирования поверхности. Угол охвата земной поверхности сканом, таким образом, меняется от минимума в середине (когда сканирующие элементы направлены в надиры) до некоторого максимума на краях. Это приводит к тому, что вдоль границ скана идет разрыв географических координат, а также к частичному перекрытию полос на краях (bow-tie effect). Одно из первых упоминаний об этом эффекте содержится в [6], в [7] дается одно из решений, но только в виде программного кода, без объяснения алгоритмов, с указанием, что можно выбрать один из алгоритмов: ближайшего соседа, билинейную или бикубическую интерполяцию. Обработка сбойных данных не описана и по-видимому не выполняется. Для первоначально алгоритма [2] такие данные не подходили, так как содержали разрывы.

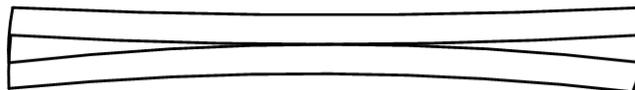


Рисунок 1. Перекрытие двух сканов из сеанса со спутника Terra (прибор MODIS) после проецирования

## 2. ОБЗОР АЛГОРИТМА

Алгоритм, использовавшийся ранее [2], был переработан и улучшен, с сохранением некоторых старых решений и применением новых. В новом варианте преобразование изображения делится на два этапа – предобработка и собственно трансформация. В процессе предобработки за один проход сразу для всего изображения вычисляются данные, которые в ином случае пришлось бы вычислять отдельно в процессе преобразования для каждого пикселя выходного изображения. За счет этого предобработка позволяет ускорить процесс трансформации. На втором этапе происходит преобразование – аппроксимация пикселей выходного изображения с адаптивным выбором алгоритма аппроксимации на основе вычисленных на этапе предобработки данных. Выходом алгоритма являются изображения, получившиеся в результате преобразования.

В качестве входных параметров алгоритм принимает:

- одномерный массив исходных изображений (слоев) одинакового размера, которые будут подвергаться преобразованию (алгоритм поддерживает преобразование нескольких слоев параллельно, что быстрее, чем преобразование каждого слоя последовательно, так как все вычисления производятся один раз на все слои, отдельно для каждого слоя происходит только конечная аппроксимация интенсивности пикселя);

- двумерный массив координат в ИСК, который соответствует по размерам исходному изображению и определяет координаты каждого пикселя в ИСК;

- функцию трансформации, которая является объектом, предоставляющим операции преобразования точки из ИСК в РСК, обратного преобразования (РСК в ИСК), а также операции определения возможны ли эти преобразования (для задания области определения и области значения). Конструктивно функция трансформации представлена в виде абстрактного класса C++, определяющего необходимый интерфейс, при этом все функции преобразования должны представляться классами, унаследованными от данного абстрактного класса. Такой унифицирующий подход делает алгоритм легко расширяемым с точки зрения задаваемого преобразования, и позволяет использовать преобразования достаточно произвольного вида;

- масштаб (задает соотношение между РСК и пиксельной результирующей системой координат (ПРСК));

На этапе предобработки вначале происходит проверка корректности исходных данных, а затем вычисление следующих данных:

- массива выходных координат, получаемого путем преобразования исходного массива координат в РСК;

- размера выходного изображения;

- инициализация и заполнение матрицы распределения и матрицы покрытия;

На втором этапе алгоритма последовательно обрабатывается каждый пиксель выходного изображения. Используя известный из матрицы покрытия локальный коэффициент сжатия, выбирается [2] оптимальный алгоритм аппроксимации (бикубический при коэффициенте сжатия менее 1.6 и по среднему – при более высоком коэффициенте сжатия).

Для аппроксимации по среднему используется матрица распределения, из которой известно, какие пиксели исходного

изображения попали после проецирования в аппроксимируемый пиксель выходного изображения. Это позволяет усреднить значение интенсивностей этих пикселей и приписать получившееся значение выходному пикселю.

При бикубической аппроксимации используется известный из матрицы покрытия ближайший пиксель исходного изображения от точки, в которую попадает аппроксимируемый пиксель выходного изображения при обратном преобразовании (из РСК в ИСК). Этот пиксель и его соседи (образующие сетку 4x4) используются для бикубической аппроксимации. При этом алгоритм учитывает то, что исходная координатная сетка (задаваемая двумерным массивом координат) не является в общем случае прямоугольной (например, сетка географических координат) и строит преобразование из ИСК в пиксельную исходную систему координат (ПИСК - система координат, где единицей измерения является пиксель и связанная с исходным растровым изображением) для нахождения смещения аппроксимируемого пикселя относительно исходной пиксельной сетки.

## 3. МАТРИЦЫ РАСПРЕДЕЛЕНИЯ И ПОКРЫТИЯ

На этапе предобработки основной задачей алгоритма является заполнение матриц распределения и покрытия. Перед заполнением этих матриц алгоритм переводит заданный на входе массив исходных координат из ИСК в РСК, одновременно находя размеры выходного изображения. Перевод координат в РСК и запись их в другой массив проводится для повышения скорости работы алгоритма (в дальнейшем такие преобразования приходится делать несколько раз на пиксель, что делает целесообразным подсчет значения один раз и его сохранение для последующего использования). Этот процесс можно представить в следующем виде:

```
preprocess_coordinate_array()
begin
    area = (0,0,0,0)
    for_each( i,j from Iarray ) do begin
        p = Iarray[i,j]
        if( canICStoRCS(p) ) then begin
            area = (ICStoRCS(p),ICStoRCS(p))
            break;
        end
    end
    for_each( i,j from Iarray ) do begin
        p = Iarray[i,j]
        if( canICStoRCS(p) ) then begin
            pR = ICStoRCS(p)
            expand(area,p)
            Rarray[i,j] = pR
        end
    end
end
```

Здесь area – прямоугольник, описывающий все точки выходного массива координат на РСК, Iarray – входной массив координат, Rarray – выходной массив координат, p, pR – точки, i, j – индексы массивов. Функция expand() расширяет прямоугольник area до таких размеров, чтобы он включал в себя точку p. Функция canICStoRCS() определяет, попадает ли точка в область определения заданного преобразования. Функция ICStoRCS() преобразовывает точку из ИСК в РСК в соответствии с заданным преобразованием. Первый цикл алгоритма необходим для инициализации area – находится

первая точка исходных координат в которой определено преобразование из ИСК в РСК и в ареа записывается прямоугольник нулевой площади вырожденный в точку. Во втором цикле этот прямоугольник расширяется до тех пор, пока не охватывает все обработанные точки.

Полученный прямоугольник, переведенный с помощью заданного масштаба в ПРСК и округленный до целых величин (пикселей), определяет размеры выходного изображения. Такой же размер имеют и матрицы распределения и покрытия.

Матрица распределения имеет следующую структуру. Изначально все ячейки матрицы пусты. Затем каждый исходный пиксель переводится из ИСК в РСК, а затем в ПРСК. Таким образом, получают координаты пикселя результирующего изображения, которым однозначно соответствует ячейка матрицы распределения, в которую записываются координаты данного исходного пикселя в ПИСК (ячейки матрицы распределения представлены как списки координат пикселей, возможно пустые). В итоге матрица представляет картину распределения пикселей исходного изображения по пикселям результирующего и позволяет определить какие пиксели исходного изображения попали в каждый конкретный пиксель результирующего.

Матрица покрытия организована иным способом. Ее ячейки также изначально пусты, но заполнение происходит по-другому. Аналогично с матрицей распределения каждый пиксель исходного изображения переводится из ИСК в ПРСК, вместе с тремя соседями, образуя ячейку. Площадь этой ячейки в ПИСК равна 1 (один квадратный пиксель), а площадь в ПРСК можно посчитать исходя из посчитанных координат. Зная эти две площади можно получить локальный коэффициент сжатия. Этот коэффициент сжатия записывается во все ячейки матрицы покрытия, которые попадают в полученный четырехугольник (образуемый четырьмя пикселями в ПРСК). Туда же записываются координаты в ПИСК обрабатываемого пикселя исходного изображения. Существует два особых случая, при которых не происходит запись в ячейку матрицы покрытия – если четырехугольник в ПРСК имеет самопересечение (особая точка в преобразовании) или если в данной ячейке уже записан локальный коэффициент сжатия и он больше, чем посчитанный (для отсека паразитных четырехугольников, получающихся когда точки лежат по разные стороны разрыва и коэффициент сжатия очень маленький).

Эта матрица, впервые введенная в данной версии алгоритма, выполняет следующие задачи, придающие алгоритму новые, отсутствующие ранее свойства:

- предоставляет значение локального коэффициента сжатия при последующей аппроксимации. Эта оценка точнее, быстрее и более плавно меняется, чем использовавшаяся в предыдущей версии алгоритма оценка по матрице распределения, где были возможны “рваные” значения коэффициента сжатия (из-за того, что на один пиксель в ПРСК могло попасть, к примеру, два пикселя ПИСК, а на соседний – ни одного);
- предоставляет хорошее приближение для поиска ближайшего пикселя к точке, куда попадает результирующий пиксель в ИСК после преобразования из ПРСК в ИСК. Ранее ближайший пиксель также приходилось искать по матрице распределения, что могло существенно повысить время работы алгоритма в случаях малого (менее 0.5) коэффициента сжатия;

- позволяет определить, попало ли на данный пиксель результирующего изображения исходное изображение. С появлением данной матрицы, можно больше не строить область, куда попадает исходное изображение на результирующем, что снимает ограничение на непрерывность исходных данных, что было одним из важных недостатков предыдущего варианта алгоритма.

#### 4. АППРОКСИМАЦИЯ ПИКСЕЛЕЙ

На втором этапе работы алгоритма происходит аппроксимация каждого пикселя выходного изображения:

```

process_pixel( prp )
begin
    if( !isFilledPixel(prp) ) then
        return
    if( getZoom(prp) >= 1.6 ) then begin
        approximateAverage(prp)
    end
    else begin
        rp = PRCStoRCS(prp)
        if( !canRCStoICS(rp) then
            return
        ip = RCStoICS(rp)
        pipAnchor = getAnchor(prp)
        if( !findCell(ip, pipAnchor, pip4, ip4) ) then
            return;
        pip = ICStoPICS(ip, pip4, ip4)
        approximateBicubical(prp, pip)
    end
end

```

В данной функции prp – обрабатываемый пиксель в координатах ПРСК, rp – пиксель в РСК, ip – пиксель в ИСК, pip – пиксель в ПИСК. pip4 и ip4 – массивы размером 4, представляющие ячейку, в ПИСК и ИСК соответственно. Функции:

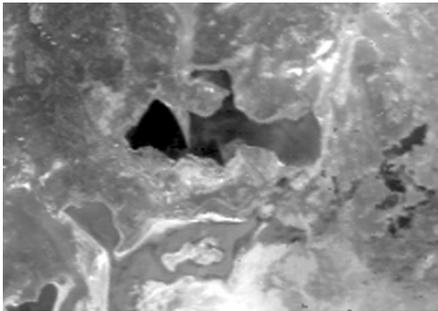
- isFilledPixel() – функция, которая определяет по матрице покрытия, попало ли на данный пиксель исходное изображение;
- getZoom() определяет из матрицы покрытия величину локального коэффициента сжатия;
- approximateAverage() – аппроксимация пикселя по среднему с помощью матрицы распределения;
- PRCStoRCS() и RCStoICS() – функции перевода из ПРСК в РСК и из РСК в ИСК соответственно;
- canRCStoICS() определяет, попадает ли точка в область определения обратного преобразования, из РСК в ИСК;
- getAnchor() возвращает хранящееся в заданной ячейке матрицы покрытия значение координат пикселя исходного изображения в ПИСК;
- findCell() использует pipAnchor для нахождения клетки исходного изображения, образованной четырьмя соседними пикселями, в которую попадает заданная точка (ip), результат помещается в массивы pip4 и ip4.
- ICStoPICS() преобразует точку из ИСК в ПИСК, путем линейной интерполяции, используя массивы pip4 и ip4. Таким образом, получают координаты аппроксимируемого пикселя в ПИСК, что позволяет проводить аппроксимацию на исходной пиксельной сетке, в отличие от предыдущей версии алгоритма, который аппроксимировал по выходной пиксель-

ной сетке, что приводило к искажениям в местах где выходная сетка координат была сильно растягнута в каком-либо одном направлении;

- approximateBicubical() – бикубическая аппроксимация.



**Рисунок 2.** Изображение со спутника Тегга. Видны разрывы на краях сканов



**Рисунок 3.** Изображение со спутника Тегга, переведенное алгоритмом трансформации в полярную стереографическую проекцию. Эффект разрывов на краях сканов скорректирован. Таким образом, для каждого пикселя выходного изображения проверяется, попало ли на него преобразуемое изображение. Если попало, то определяется локальный коэффициент сжатия, и в зависимости от него используется либо метод аппроксимации по среднему, либо бикубическая аппроксимация.

## 5. РЕАЛИЗАЦИЯ И ИСПОЛЬЗОВАНИЕ

Представленный алгоритм был реализован в виде библиотеки классов C++ и испытан на произвольных преобразованиях вида растяжения, поворота и более сложных. Был реализован перевод сеансов со спутника Тегга (MODIS) с одновременным (за один проход) устранением эффекта наложения полос из-за расширения угла зрения камеры на краях скана (bow-tie effect [6,7]). За счет того, что полосы частично перекрываются, возникает избыточность информации, что позволяет алгоритму частично восстанавливать зашумленные при приеме сеанса данные. В старом варианте алгоритма такое преобразование было невозможным без разделения исходного изображения на сканы и проецирования отдельных сканов. Модуль проецирования с переработанным алгоритмом трансформации предназначен для замены уже встроенного в программно-аппаратный комплекс AlcorSat [8] модуля, который использует предыдущий вариант алгоритма. Предложенный

алгоритм также заменил старый [2] в программной библиотеке для накопления, хранения и обработки карт местности, формируемых из спутниковых изображений [9].

## 6. ССЫЛКИ

- [1] Прэтт У. Цифровая обработка изображений: Пер. с англ. – М.: Мир, 1982. – Кн.1 – 312 с., ил., Кн.2 – 480 с., ил
- [2] Nepomnyaschy P.V., Yurin D.V. Satellite Images Transformation to Geographical Projection with Adaptive Choice of Optimal Approximation Method According to Local Compression Coefficient. // In Conference Proceedings. 12-th International Conference on Computer Graphics GraphiCon'2002 –P. 299–304. Nizhny Novgorod, September 16-21, 2002.
- [3] Непомнящий П.В., Хельвас А.В., Юрин Д.В. Универсальный алгоритм перерастрирования изображений с автоматическим выбором оптимального способа аппроксимации. / В сб. «Обработка информации и моделирование». -М.:МФТИ, -2002, -С. 56-65, 355 стр. УДК 519.687, ISBN-5-7417-0184-1.
- [4] Direct Readout Portal.  
[http://directreadout.gsfc.nasa.gov/documents/drl\\_docs\\_main.cfm](http://directreadout.gsfc.nasa.gov/documents/drl_docs_main.cfm).
- [5] MODIS command, telemetry, science and engineering description Santa Barbara Research Center, A Subaidiary of Hughes Aircraft Company, Goleta, California  
[http://directreadout.gsfc.nasa.gov/documents/satellite\\_gen/MODIS\\_UG.pdf](http://directreadout.gsfc.nasa.gov/documents/satellite_gen/MODIS_UG.pdf).
- [6] Nishihama M., Wolfe R., Solomon D., Patt F., Blanchette J., Fleig A., Masuoka E., Level 1A Earth Location, version 3.0 08/26/1997, Algorithm Theoretical Basis Document, ATBD-MOD-28 [http://modis.gsfc.nasa.gov/data/atbd/level1\\_atbd.html](http://modis.gsfc.nasa.gov/data/atbd/level1_atbd.html),  
[http://modis.gsfc.nasa.gov/data/atbd/atbd\\_mod28\\_v3.pdf](http://modis.gsfc.nasa.gov/data/atbd/atbd_mod28_v3.pdf).
- [7] MODIS Swath Reprojection Tool (MRTSwath) Overview.  
<http://lpdaac.usgs.gov/landdaac/tools/mrtswath/about.asp>.
- [8] Официальный сайт ФГУП НПП ОПТЭКС  
[http://www.optecs.ru/index\\_e.htm](http://www.optecs.ru/index_e.htm).
- [9] Nepomnyaschy P.V. Current maps update on a basis of regularly incoming images of earth surface. // In Conference Proceedings. 13-th International Conference on Computer Graphics GraphiCon'2003 –P. 208–212. Moscow, September 5-10, 2003.

## Об авторе

Непомнящий Павел Владимирович, ЦОС&ВТ МФТИ,  
[nrv@cos.ru](mailto:nrv@cos.ru)